

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP

TELEPHONE: (310) 207-3800

INTELLECTUAL PROPERTY LAW
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040FACSIMILE: (310) 820-5988
(310) 820-5720

FACSIMILE COVER SHEET

Deliver to: Examiner Taha, Shaq _____ Art Group: _____ 2122

Company Name: USPTO _____

Facsimile No.: 571-270-2921. _____ Date: 02/05/01

From: Jonathan S. Miller, Reg. No. 48,534

Our Docket No.: 6570P011 Number of pages 8 including this sheet.

Application: 10/750,280 Filing Date: 12/30/2003

Subject

Remarks

Please find enclosed a Proposed Examiner's Interview for Application 10/750,280

Confidentiality Note: The documents accompanying this facsimile transmission contain information from the law firm of Blakely, Sokoloff, Taylor & Zafman which is confidential or privileged. The information is intended to be for the use of the individual or entity named on this transmission sheet. If you are not the intended recipient, be aware that any disclosure, copying, distribution or use of the contents of this faxed information is prohibited. If you have received this facsimile in error, please notify us by telephone immediately so that we can arrange for the retrieval of the original documents at no cost to you.

If you do not receive all the pages, or if there is any difficulty in receiving, please call: (310) 207-3800 and ask for Jessica M. Huester.

Attorney's Docket No. 6570P011

Proposed First Examiner's Interview for Application 10/750,280**ATTN: Examiner Taha, Shaq**

The Applicant submits a proposed agenda herein for an Examiner's interview in regard to the Office Action mailed December 23, 2008. The Examiner has proposed a tentative interview time of **3:00 PM (EST) on February 19, 2009**. Please contact **Dimitri Kirimis** at phone number (310) 500-4751 with any questions regarding this proposed agenda and to confirm the interview time.

The Applicant is willing to discuss the prior art in detail with respect to the proposed claim attachments. In turn, the Applicant respectfully requests that the Examiner be prepared, at a minimum, to discuss the sections of the prior art cited in the following sections, along with the following questions. The purpose of this agenda is to set forth issues that need to be addressed to further prosecution of the application. In other words, the Examiner can expect the scope of the interview to fall within what is set forth in the agenda without being concerned that the Applicant is seeking to address other matters not mentioned herein.

In the Office Action, Independent claims 1, 7, 15, 21, and 25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent Publication No. 2004/0146056 by Martin et al. (hereinafter "Martin") in view of U.S. Patent No. 7,114,170 issued to Harris et al. (hereinafter "Harris").

In regard to claim 1, the Applicant proposes an amendment which includes "maintaining a list of services, the list of services includes processes and tasks performed by the Java-based server nodes; and maintaining a list of services, the list of services includes processes and tasks performed by the non-Java-based server nodes." (emphasis added). These proposed amendments are supported, for example, by paragraph [0020] of the Specification. The combination of Martin and Harris does not disclose maintaining a list of services and tasks performed by a server. Martin discloses maintaining a list of routes. See Martin, Abstract. However, routes are not processes or tasks. Thus, Martin does not disclose maintaining a list of processes and services. Further, Harris does not cure the deficiencies of Martin, because there is no disclosure in Harris of maintaining a list of any data.

In light of the preceding statements, would the proposed amendments overcome the § 103 rejection of claim 1 based on the combination of Martin and Harris?

In regard to claim 7, the Applicant proposes an amendment which includes "an enqueue server coupled between the first and second instances to provide central locking services to lock access to resources in the system." These proposed amendments are supported, for example, by paragraph [0018] of the Specification. The combination of Martin and Harris fails to disclose an enqueue server to provide locking services, because these references do not discuss providing locking services for their respective systems. Thus, the combination of Martin and Harris does not disclose "an enqueue server to provide central locking services to lock access to resources in the system."

In light of the preceding statements, would the proposed amendments overcome the § 103 rejection of claim 7 based on the combination of Martin and Harris?

Although the below discussion is focused on the elements of claim 15, these reasons are applicable to independent claim 21 as well because this claim recites analogous limitations to those in claim 15.

In regard to claim 15, the Applicant proposes an amendment which includes "a controller to transfer packets between the non-Java-based server nodes and the Java-based server nodes, the controller to ensure the packets are received by a destination server node by resending the packets if a confirmation of receipt has not been received from the destination server node" (emphasis added). These proposed amendments are supported, for example, by paragraph [0030] of the Specification. Martin discloses decoding a packet at a packet switch. See Martin, Claim 8. The packet switch decodes the time in the packet header and feeds it to the updating means. See Id. The updating means subsequently stores the latency for packet transmission through a designated route. See Martin, Claim 7. However, Martin does not disclose resending packets if a confirmation of receipt has not been received from the destination server node, because there is no discussion of resending a packet. Further, Harris does not cure the deficiencies of Martin.

In light of the preceding statements, would the proposed amendments overcome the § 103 rejection of claims 15 and 21 based on the combination of Martin and Harris?

In regard to claim 25, the Applicant proposes an amendment which includes "means for maintaining a list of services, the list of services includes processes and tasks performed by the Java-based server nodes; and means for sending notification of a status of each of the listed services to the Java-based server nodes, the notification indicates whether the service is running or stopped" (emphasis added). These proposed amendments are supported, for example, by paragraph [0022] of the Specification. Martin discloses maintaining a routing table that records the latency of routes. See Martin, Abstract. However, this routing table does not store a list of services including processes and tasks, for at least the same reasons discussed above in relation to the proposed amendment of claim 1. Further, storing latency information does not disclose sending a notification of whether a service is running or stopped, because Martin does not disclose sending a notification related to the status of a service as a result of storing latency data. Moreover, latency of a route does not disclose whether a service is running or stopped, because the latency reflects the delay in an associated path and it is not directly associated with whether a service is running or is stopped. Thus, Martin does not disclose these elements of amended claim 25. Further, Harris does not cure the deficiencies of Martin.

In light of the preceding statements, would the proposed amendments overcome the § 103 rejection of claim 25 based on the combination of Martin and Harris?

Further, the Applicant presents a proposed new claim 31. This proposed new claim is supported by original claim 7 and paragraphs [0020]-[0024] of the Specification as filed. Claim 31 recites "a first repository to maintain a list of processes and tasks performed by the plurality of non-Java based server nodes...a second repository to maintain a list of processes and tasks performed by the plurality of Java based server nodes...and a third repository to maintain a list of assigned service identifications and their corresponding service names." The combination of Martin and Harris do not disclose a structure for maintaining a list of processes and tasks as recited in claim 31, because these references do not discuss a structure for maintaining a list of "processes and tasks."

In light of the preceding statements, would the proposed new claim 31 overcome the § 103 rejection based on the combination of Martin and Harris?

PROPOSED CLAIM AMENDMENTS

1. (Currently Amended) A method comprising:
 - establishing communication between a plurality of non-Java-based server nodes of a first instance and a plurality of Java-based server nodes of a second instance via an intermediate server;
 - generating a packet, on one of the non-Java-based server nodes, to be transmitted from one of the non-Java-based server nodes to one of the Java-based server nodes;
 - specifying in a header of the packet an address of a destination Java-based server node and information that indicates that the packet is generated by one of the non-Java-based server nodes;
 - forwarding the packet to the intermediate server from the one of the non-Java-based server nodes;
 - forwarding the packet to the destination Java-based server node from the intermediate server based on the address provided in the header of the packet by synchronizing the packet header such that it can be decoded by a destination server;
 - maintaining a list of services, the list of services includes processes and tasks performed by the Java-based server nodes; and
 - maintaining a list of services, the list of services includes processes and tasks performed by the non-Java-based server nodes.
7. (Currently Amended) A system comprising:
 - a first instance including a plurality of non-Java-based server nodes, each of the non-Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from one of the non-Java-based server nodes;
 - a second instance including a plurality of Java-based server nodes, each of the Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from one of the Java-based server nodes; and

a message server coupled between the first and second instances to establish communication there between the first instance and the second instance by synchronizing each packet header such that it can be decoded by a destination server; and

an enqueue server coupled between the first and second instances to provide central locking services to lock access to resources in the system.

15. (Currently Amended) A message server comprising:

a first communication interface to establish communication with a plurality of non-Java-based server nodes;

a second communication interface to establish communication with a plurality of Java-based server nodes; and

a controller to transfer packets between the non-Java-based server nodes and the Java-based server nodes, the controller to ensure the packets are received by a destination server node by resending the packets if a confirmation of receipt has not been received from the destination server node.

21. (Currently Amended) A machine-readable medium that provides instructions, which when executed by a processor cause the processor to perform operations comprising:

establishing communication with a plurality of non-Java-based server nodes;

establishing communication with a plurality of Java-based server nodes;

transferring packets between the non-Java-based server nodes and the Java-based server nodes; and

ensuring the packets are received by a destination server node by resending the packets if a confirmation of receipt has not been received from a destination server node.

25. (Currently Amended) A system comprising:

means for generating a packet such that a header of the packet specifies an address of a destination Java-based server node;

means for indicating that the packet is generated by a non-Java based server node;

means for forwarding the packet to intermediate communication means from the one of the non-Java-based server nodes; and

means for forwarding the packet to the destination Java-based server node from the intermediate communication means based on the destination address provided in the header of the packet;

means for maintaining a list of services, the list of services includes processes and tasks performed by the Java-based server nodes; and

means for sending notification of a status of each of the listed services to the Java-based server nodes, the notification indicates whether the service is running or stopped.

31. (New) A system comprising:

a first instance including a plurality of non-Java-based server nodes, each of the non-Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from one of the plurality of non-Java-based server nodes;

a second instance including a plurality of Java-based server nodes, each of the Java-based server nodes executing software instructions to attach a header to a body of a packet, the header including information to specify that the packet originated from one of the plurality of Java-based server nodes; and

a message server coupled between the first and second instances to establish communication there between the first instance and the second instance by synchronizing each packet header such that it can be decoded by a destination server, the message server comprising:

a first repository to maintain a list of processes and tasks performed by the plurality of non-Java based server nodes, the first repository includes a plurality of rows each associated with a service performed by a non-Java based server and a plurality of columns which identify attributes associated with each service, the attributes include a server identification to indicate a server performing the service, a service mask to identify the type of service, and a status to indicate the status of the service;

a second repository to maintain a list of processes and tasks performed by the plurality of Java based server nodes, the second repository includes a plurality of rows each associated with a service performed by a Java based server and a plurality of columns which identify attributes associated with each service, the attributes include a

server identification to indicate a server performing the service, a service mask to identify the type of service, and a status to indicate the status of the service; and

a third repository to maintain a list of assigned service identifications and their corresponding service names, the third repository includes a plurality of rows each associated with a service and a plurality of columns which identify attributes associated with each service, the attributes include a service name to indicate the name of the service and an assigned service mask to indicate the type of service.